A GRAPHICS INTERFACE TO LARGE, SHARED DATABASES
A Summary Paper

Mark Friedell
Computer Corporation of America and
Case Western Reserve University

Jane Barnett and David Kramlich
Computer Corporation of America

ABSTRACT

We begin by reviewing spatial data management -- the technique of organizing and accessing information via its graphical representation in in a spatial framework. We describe an operational prototype system that exceeds the capabilities of other spatial data management systems in two ways: (1) the graphical presentation of data is tailored to the user's identity and task, and (2) the system has the capacity for very large, shared databases. These capabilities are possible because the system dynamically generates its graphics environment as the user moves through space, poses new database queries, and changes viewing task. Our technique for dynamically generating the graphics environment relies on semantic modeling of the underlying database, modeling of user contexts, and the basic but direct use of knowledge about design layout and the utility of pictures. We briefly describe our current research efforts to extend this knowledge-based approach.

KEYWORDS: computer graphics; database management; database semantics; graphical interface; graphics environment synthesis; man-machine interfaces; query menu; semantic modeling; spatial data management

CR Categories: 3.69, 3.70, 8.2

1. Introduction

Spatial data management is the technique of graphically organizing and presenting information. Prototype spatial data management systems have been built that successfully exploit computer graphics as a database access medium [DONELSON] [HEROT ET AL]. However, this success is limited to a small number of applications using relatively small databases.

This researoh was supported by the U.S. Naval Electronics Systems Command under contract N00039-80-C-040 and by the U.S. Defense Advanced Research Projects Agency under contract N00014-81-C-0456. The views and conclusions contained in this document are not necessarily those of the U.S. Government.

These limitations are principally due to the large investment required to generate and store graphics environments. In general, the costs of graphically presenting a large database are prohibitive. Maintaining several different graphic representations of a single database -- each tailored to a different task -- is also impractical. Hence, spatial data management has been applied only to tasks that could benefit from the "typical" representation of a relatively small database.

This paper reports on the development of the VIEW system, a newly operational, spatial data management system that addresses the environment synthesis and management problems. VIEW dynamically defines and generates its graphics environment as the user moves through space, poses new database queries, and changes his viewing task. This technology allows VIEW to graphically present very large databases and to present them in several different ways, as required by different user tasks.

2. The Spatial Data Management Concept

Spatial data management is the technique of organizing and presenting information by presenting and positioning it graphically. In the VIEW system, data are viewed through color displays. The displays show flat "data surfaces" on which pictorial representations of the data are arranged. The data surfaces are referred to as Information spaces or Ispaces. The pictorial representations of data are termed "icons." For a typical, entity-oriented database, each icon represents one entity.

The user can traverse an Ispace and "zoom into" an icon to obtain greater detail. This approach permits many types of questions to be answered without requiring the use of a keyboard and a conventional query language.

Spatial data management is motivated by the needs of a growing community of people who want to access information through a database management system (DBMS) but are not trained in the use of such systems. A database viewed through a spatial data management system is more accessible and its structure is more apparent than when viewed through a conventional DBMS. Users of conventional DBMSs can access data only by asking questions in a formal query language. In contrast, users of a spatial data management system benefit from the ability to access computer-resident information while retaining a familiar, visual orientation.

By presenting information in a natural, spatial framework, spatial data management encourages browsing and requires less prior knowledge of the contents and structure of the database. A user can find the information he needs without having to specify it precisely or know exactly where in the database it is stored. Users can easily organize, locate, and handle a great deal of information of different types.

3. The View System

The View System includes two primary components: the Answer Space Generator and the View Generator. The Answer Space Generator interacts with the user through a graphical, query interface to produce descriptions of the user's identity, overall task goals, and specific database queries. The View Generator is responsible for using these descriptions to create Ispaces and to respond to the user's requests for motion though an Ispace.

3.1 The Answer Space Generator

The aim of the Answer Space Generator is to provide the user with a spatially oriented model of the data contained in the database and to allow the specification of queries without a formal query language. All knowledge about the nature of the data contained in the database is consolidated into a unified "conceptual model." This model is based on an object-oriented, knowledge representation in the form of a semantic network. Each node or "concept" in the network represents a generic definition of a set of data entities in the database. The description of each concept includes the concept's name, attributes, graphical properties, and relationships to other concepts in the network. The conceptual model also includes information about how data should be retrieved from the database to instantiate the concepts.

The Answer Space Generator allows the user to move about in the conceptual model and to specify queries in terms of the entities described in this network. Most of these interactions are accomplished by simply pointing to the touch-sensitive screens mounted on two color displays. One display shows the user an overview of the conceptual model, known as the "concept map." This map shows the entities contained in the conceptual model and the relationships among them. It also records, by means of highlights, the entities specified in the current query. The concept map continually provides a clear picture of the user's location and status with respect to the system.

The second display presents a more-detailed view of the user's location on the concept map and a menu of functions. The central portion of this screen shows the concept currently in "focus," its attributes, and the concepts closely related to it. The bottom of the screen is reserved for the function buttons, which allow the user to change the concept in focus, select a concept for viewing, qualify a concept, initiate data viewing, or exit the system.

When the user completes a query, the Answer Space Generator retrieves the appropriate data, creates a semantic description of the Ispace, and passes the data and the description to the View Generator. The Ispace description includes the names of the data entities retrieved for display and information about each entity type, range of values, and number of unique values. Each entity is ranked based on its "importance" to the query, the task goal, and the user. This approach allows displays to be dynamically tailored to the user's context.

3.2 The View Generator

The View Generator treats the Ispace as a virtual graphics environment. While the logical contents and layout of the Ispace are completely defined, only a small region surrounding (and including) the screen is instantiated. The instantiated area exists in the "Ispace cache."

The virtual Ispace approach has two important advantages. First, storage for the entire Ispace is not needed. The conventional approach of storing the entire Ispace places an upper bound on Ispace size. This constraint is effectively eliminated in the View Generator.

The second advantage relates to "Ispace integrity maintenance" -- the problem of maintaining consistency between the representation of information in the Ispace and its representation in the underlying, alphanumeric database. Inconsistencies can occur if the alphanumeric database is updated after the Ispace is generated. Using a virtual Ispace, only those updates affecting the appearance of the small Ispace cache need be considered.

The View Generator produces an Ispace in two phases. The first phase is formulating a logical Ispace layout. The second phase is creating the appropriate icons to fill the cache, as the user moves through Ispace.

The architecture of the View Generator is shown in Figure 3.1. The major modules are discussed below.

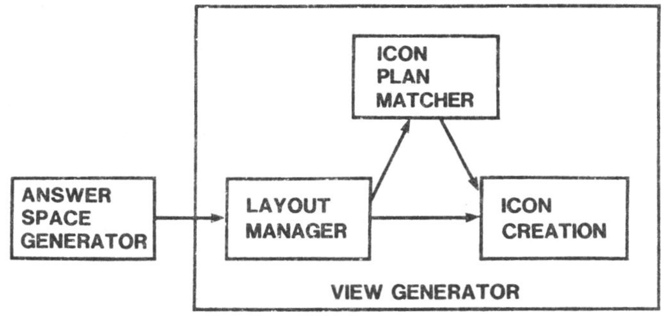

Figure 3.1 Architecture of the View Generator

The Layout Manager is responsible for two tasks within the View Generator: formulating the layout of icons within an Ispace and managing the Ispace cache in response to user motion through the Ispace. In its role as layout designer, the Layout Manager serves as the interface between the Answer Space Generator and the View Generator. The Layout Manager receives a description of the query posed by the user from the Answer Space Generator.

Using this description of the data and a simple set of heuristics for arranging icons on the basis of data distributions and data types, the Layout Manager formulates a layout plan for the Ispace. The Layout Manager then requests that the data be retrieved from the DBMS. For each entity retrieved, the Layout Manager calculates the location in the Ispace of its corresponding icon (based on its layout plan) and allocates space for it.

The View Generator relies on a repertoire of <u>icon plans</u> to specify how icons should be created. Associated with each plan is an applicability pattern. This pattern describes the ideal situation in which the plan ought to be used.

The applicability pattern is couched in terms of user identity, user task, the entity to be presented, the attributes to be presented, the relative importance of each attribute, and the graphic properties of resulting icons.

Icon plans are composed in a high-level, graphics-oriented, plan language that may be used by a human expert or (in the future) by an automatic plan synthesizer (see section 4). A plan language compiler is used to prepare the plan for efficient interpretation by the View Generator.

The Icon Plan Matcher chooses the best plan to use for the current icon creation task. It makes this determination by incrementally generalizing the task description until it matches a plan's applicability pattern.

4. Future Research

Much of VIEW's ability to generate Ispaces dynamically can be credited to the use of embedded knowledge of database semantics, user identity, user task, and the appropriateness and utility of different types of images. Knowledge-engineering techniques such as these are relatively new to computer graphics. Only a few examples can be readily found in the literature [KAHN] [BROWN & CHANDRASEKARAN] [GREENFELD & YONKE].

Plans are underway at Computer Corporation of America to fully automate the synthesis of icon plans through a knowledge-engineering approach. This work involves techniques specially developed for computer graphics applications [FRIEDELL]. We hope that our current research, in conjunction with the work reported here, will result in a technology for fully automatic environment synthesis.

In the immediate future, we plan to apply the techniques described in this paper to the problem of automating the production of animated illustrations of computer processes. This work will be part of a long-term effort to develop a program visualization system -- a system for graphically examining and manipulating dynamic representations of large computer programs [HEROT, BROWN, ET AL].

5. References

[BROWN & CHANDRASEKARAN]
Brown, D. and Chandrasekaran, B. "Design Considerations for Picture Production in a Natural Language Graphics System." Computer Graphics, Vol. 15, No. 2.

[DONELSON]
Donelson, W. "Spatial Management of Information." Proceedings of ACM Siggraph '78, Atlanta, Georgia.

[FRIEDELL]
Friedell, M. "A Computational Model for Graphics Environment Synthesis." Forthcoming Ph.D. Dissertation, Department of Computer Engineering and Science, Case Western Reserve University, Cleveland, Ohio.

[GREENFELD & YONKE]
Greenfeld, N. and Yonke, M. "An Information Presentation System for Decision Makers." Artificial Intelligence Department, Bolt, Beranek, and Newman, Inc., Cambridge, Massachusetts.

[HEROT ET AL]
Herot, C.; Carling, R.; Friedell, M.; and Kramlich, D. "A Prototype Spatial Data Management System." Proceedings of ACM Siggraph '80, Seattle, Washington.

[HEROT, BROWN, ET AL]
Herot, C.; Brown, G.; Carling, R.; Friedell, M.; Kramlich, D.; and Baecker, R. "An Integrated Environment for Program Visualization." Proceedings of the IFIP WG 8.1 Working Conference on Automated Tools, New Orleans, Louisiana, January, 1982 (to appear).

[KAHN]
Kahn, K. "Creation of Computer Animation from Story Descriptions." A.I. Technical Report 540 (Ph.D. Dissertation), Massachusetts Institute of Technology, Cambridge, Massachusetts.